

In the Claims:

Please amend claims 1, 9, 15, and 21 and add new claims 29 and 30. All claims in the application are reproduced below.

1. (Currently Amended) An interactive tool for manipulating at least one deployment descriptor, comprising:

a first user interface capable of rendering a hierarchical representation of the at least one deployment descriptor, wherein a component of the representation can be selected by a user;

a second user interface capable of rendering a user-editable representation of the selected component; and

a builder component capable of organizing deployment information from all of an application's deployment descriptors into a logical hierarchy of resources, wherein the hierarchical representation of the at least one deployment descriptor includes a logical representation of application resources, wherein each one of the at least one deployment descriptor describes run-time parameters particular to a given web server implementation, configuration information for application objects.

2. (Original) The interactive tool of claim 1, further comprising:

a third user interface capable of rendering an error message.

3. (Original) The interactive tool of claim 2 wherein:

user selection of the error message can cause the second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

4. (Original) The interactive tool of claim 1, further comprising:

a parser capable of generating a representation of the at least one deployment descriptor;

a generator capable of creating the at least one deployment descriptor; and
a validator capable of validating the at least one deployment descriptor.

5. (Original) The interactive tool of claim 4 wherein:
the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.
6. (Previously Presented) The interactive tool of claim 1, wherein:
the builder component is further capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.
7. (Original) The interactive tool of claim 1 wherein:
the hierarchical representation can include information pertaining to at least one of: a Java™ archive (JAR), a Web Archive (WAR), an Enterprise Archive (EAR), and a Java™ Connector Architecture Component (RAR).
8. (Original) The interactive tool of claim 1 wherein:
the at least one deployment descriptor can be expressed as an Extensible Markup Language document.
9. (Currently Amended) An interactive tool for manipulating at least one deployment descriptor, comprising:
a first user interface capable of rendering a hierarchical representation of the at least one deployment descriptor, wherein a component of the representation can be selected by a user;
a second user interface capable of rendering a user-editable representation of the selected component;

a third user interface capable of rendering an error message;

a builder component capable of organizing deployment information from all of an application's deployment descriptors into a logical hierarchy of resources, wherein the hierarchical representation of the at least one deployment descriptor includes a logical representation of application resources, wherein each one of the at least one deployment descriptor describes run-time parameters particular to a given web server implementation, configuration information for application objects; and

wherein user selection of the error message in the third user interface can cause the second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

10. (Original) The interactive tool of claim 9, further comprising:
a parser capable of generating a representation of the at least one deployment descriptor;
a generator capable of creating the at least one deployment descriptor; and
a validator capable of validating the at least one deployment descriptor.
11. (Original) The interactive tool of claim 10 wherein:
the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.
12. (Original) The interactive tool of claim 9, further comprising:
a builder component capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.
13. (Original) The interactive tool of claim 9 wherein:

the hierarchical representation can include information pertaining to at least one of: a Java™ archive (JAR), a Web Archive (WAR), an Enterprise Archive (EAR), and a Java™ Connector Architecture Component (RAR).

14. (Original) The interactive tool of claim 9 wherein:

the at least one deployment descriptor can be expressed as an Extensible Markup Language document.

15. (Currently Amended) A method for providing an interactive tool for manipulating at least one deployment descriptor, comprising:

providing a first user interface capable of rendering a hierarchical representation of the at least one deployment descriptor, wherein a component of the representation can be selected by a user;

providing a second user interface capable of rendering a user-editable representation of the selected component;

providing a third user interface capable of rendering an error message;
organizing deployment information from all of an application's deployment descriptors into a logical hierarchy of resources, wherein the hierarchical representation of the at least one deployment descriptor includes a logical representation of application resources, wherein each one of the at least one deployment descriptor describes run-time parameters particular to a given web server implementation, configuration information for application objects; and

wherein user selection of the error message in the third user interface can cause the second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

16. (Original) The method of claim 15, further comprising:

providing a parser capable of generating a representation of the at least one deployment descriptor;

providing a generator capable of creating the at least one deployment descriptor; and
providing a validator capable of validating the at least one deployment descriptor.

17. (Original) The method of claim 16 wherein:
the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.
18. (Original) The method of claim 15, further comprising:
providing a builder component capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.
19. (Original) The method of claim 15 wherein:
the hierarchical representation can include information pertaining to at least one of: a Java™ archive (JAR), a Web Archive (WAR), an Enterprise Archive (EAR), and a Java™ Connector Architecture Component (RAR).
20. (Original) The method of claim 15 wherein:
the at least one deployment descriptor can be expressed as an Extensible Markup Language document.
21. (Currently Amended) A machine readable medium having instructions stored thereon that when executed by a processor cause a system to:

provide a first user interface capable of rendering a hierarchical representation of the at least one deployment descriptor, wherein a component of the representation can be selected by a user;

provide a second user interface capable of rendering a user-editable representation of the selected component;

provide a third user interface capable of rendering an error message; and

organize deployment information from all of an application's deployment descriptors into a logical hierarchy of resources, wherein the hierarchical representation of the at least one deployment descriptor includes a logical representation of application resources, wherein each one of the at least one deployment descriptor describes run-time parameters particular to a given web server implementation, configuration information for application objects.

22. (Original) The machine readable medium of claim 21 wherein:

user selection of the error message in the third user interface can cause the second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

23. (Original) The machine readable medium of claim 21, further comprising instructions that when executed cause the system to:

provide a parser capable of generating a representation of the at least one deployment descriptor;

provide a generator capable of creating the at least one deployment descriptor; and

provide a validator capable of validating the at least one deployment descriptor.

24. (Original) The machine readable medium of claim 23 wherein:

the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.

25. (Original) The machine readable medium of claim 21, further comprising instructions that when executed cause the system to:

provide a builder component capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.

26. (Original) The machine readable medium of claim 21 wherein:

the hierarchical representation can include information pertaining to at least one of: a Java™ archive (JAR), a Web Archive (WAR), an Enterprise Archive (EAR), and a Java™ Connector Architecture Component (RAR).

27. (Original) The machine readable medium of claim 21 wherein:

the at least one deployment descriptor can be expressed as an Extensible Markup Language document.

28. (Previously Presented) The interactive tool of claim 1, wherein:

the interactive tool is capable of automatically repairing a first deployment descriptor of the at least one deployment descriptors if the first deployment descriptor is defective.

29. (New) The interactive tool of claim 1, wherein:

the builder component is further capable of create a tree data structure that embodies hierarchical relationships of nested XML statements.

30. (New) The interactive tool of claim 1, wherein:

the builder component is further capable of:

creating a master tree data structure that represents the present state of all deployment descriptor files;

invoking a generator to create a new tree data structure that represents deployment descriptor information based on the current state of source files in an application's project directory;

comparing the master tree data structure with the new tree data structure;
and

refreshing the master tree data structure based on the new tree data structure.